



图文 01 一个真实电商订单系统的整体架构、业务流程及负载情

3572 人次阅读 2019-10-14 08:10:56

[详情](#) [评论](#)

一个真实电商订单系统的整体架构、业务流程及负载情况

石杉老哥重磅力作：《互联网java工程师面试突击》（第3季）【强烈推荐】：



全程真题驱动，精研Java面试中**6大专题的高频考点**，从面试官的角度剖析面试

(点击下方蓝字试听)

[《互联网Java工程师面试突击》（第3季）](#)

正文开始：

1、专栏主人公介绍

小猛：应届毕业Java工程师，刚刚通过校招进入一家互联网公司工作，一个职场新人，对很多技术都不是太熟悉，还需要有资深的工程师带着，讲解系统业务和架构，同时安排和布置任务给他

明哥：10年工作经验的Java架构师，常年在国内各个一线互联网公司工作，有丰富的互联网公司的核心系统的设计、开发与运维经验，目前独立带领一个团队负责公司的订单系统，小猛就是他团队中的校招应届生，由他亲自带。

说明：这个专栏会从较为真实的职场环境出发，以两个主人公在工作中的一些对话作为核心的主体内容，带着大家去熟悉一个电商订单系统的架构以及各种技术痛点。

然后会由主人公之一的明哥带着职场新人小猛去熟悉消息中间件技术，为他布置各种任务去完成系统中的一些架构优化的工作。

相信这样的一种方式，会让大家学习起来更有代入感，而且更加有趣味性。

2、一个订单系统的业务流程

小猛从学校毕业后正式入职这家公司已经有1周了，在第一周里，主要是熟悉一下公司的开发环境、开发规范，跟团队里的同事熟悉熟悉，参加一下公司统一为新人提供的入职培训，同时参加团队的一些会议，熟悉一下工作方式。

一周过后，小猛就要开始正式接手一些实际的开发任务了。今天团队leader明哥带着小猛找了一个小会议室，打算给小猛进行为期几天的1对1培训

而培训的目的，主要是让小猛熟悉一下团队负责的订单系统的核心业务流程、整体架构设计以及当前系统面临的一些技术上的难点和痛点。

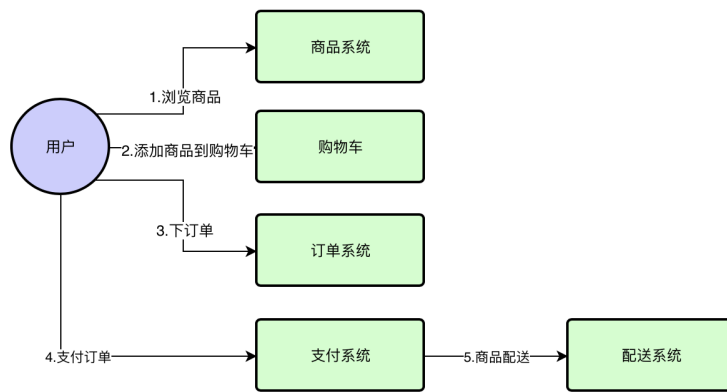
明哥：小猛，今天我来给你讲讲咱们团队负责的订单系统的业务流程以及整体架构。

小猛：好的，明哥，没问题，我会仔细听做好笔记的。

接着明哥就开始在白板上边画图，一边讲起公司订单系统的业务流程。

明哥：订单系统可以说是公司最核心的业务系统之一，因为用户要在公司的APP中购买商品，首先会在APP中浏览各种各样的商品

接着就会将喜欢的商品加入到购物车中，然后下订单进行支付后就可以购买这些商品了，你看我下面画的这个图：



所以在这个电商购物的流程中，订单系统是其中非常关键的一环，而我们团队就是负责整个订单系统的。

那订单系统在下订单这个核心的业务流程中，他自身的业务流程又是怎样的呢？

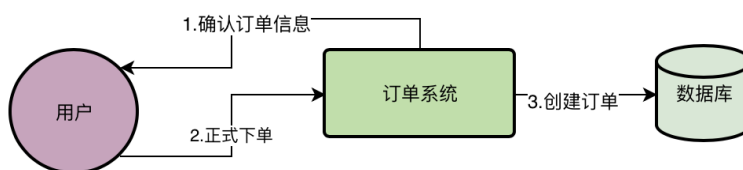
简单来说，当用户对购物车中选中的一批商品确认下单的时候，会先出来一个确认订单的界面

用户得先确认这个订单中的商品、价格、运费无误，而且在这个过程中可以选择是否要使用优惠券、促销活动的。

另外，用户还应该在这个界面中确认自己的快递方式，收件地址，是否要开发票以及发票的抬头是什么。

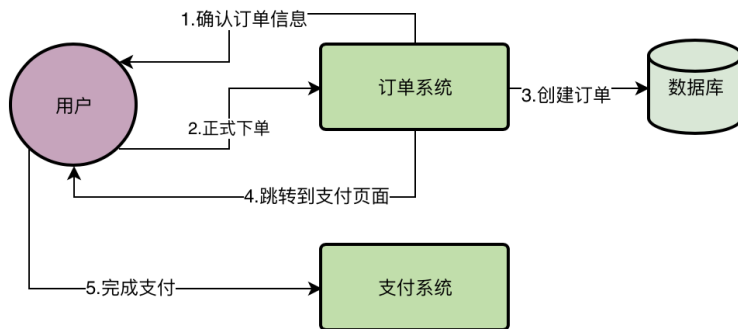
当用户完成这些信息确认之后，就可以确定下单。

此时我们的订单系统最核心的一个环节就出现了，就是要根据APP端传递过来的种种信息，完成订单的创建，此时需要在数据库中创建对应的订单记录，整个过程，就像下面这个图一样：



接着当你正式确认下单之后，除了在数据库中创建这个订单之外，还会跳转到支付界面，让你通过选择好的支付方式完成这个订单的支付

比如跳转到支付宝或者微信，让你在支付宝或者微信中完成支付，看我下面的图：



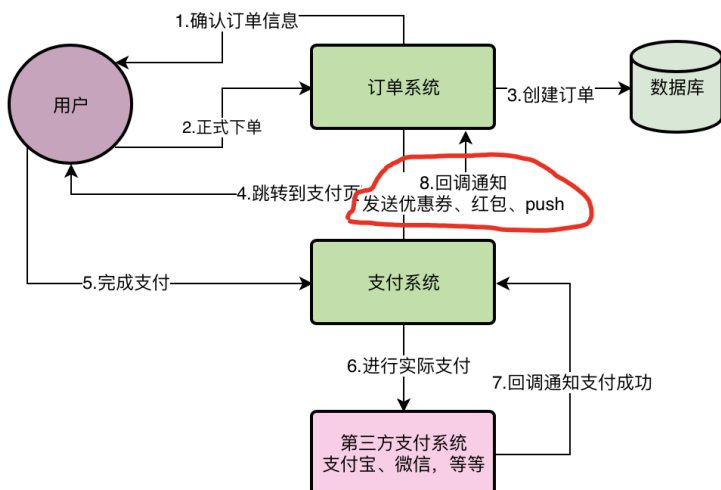
在完成了支付之后，一般来说，支付宝或者微信之类的支付系统，会反过来回调我们的一个接口，通知我们本次支付已经成功。

当我们收到支付成功的通知之后，就需要安排给用户进行配送发货。除此之外，我们的订单系统需要负责给用户发放优惠券一类的东西。

因为一般电商APP都经常会做一些鼓励用户购买的活动，比如你购买之后送一些优惠券，下次购买可以抵用5块钱，或者给你发一个几块钱的现金红包。

此外，还会给你发送一个push推送，通知你支付成功准备发货，这个推送很多时候是通过短信通知的。

我们看下面的图，在下面的图中就有支付完订单之后要做的一些事情：



明哥说到这里，休息了一下，问道：怎么样小猛，对上面说的订单系统的核心业务流程都搞明白了吗？

小猛：嗯嗯，差不多明白了，因为平时就经常会用一些电商APP去进行购物，所以对整个业务流程结合自己平时的使用体验，还是很容易搞懂的。

3、订单系统的非核心业务流程

明哥点点头：好的，看来上面的核心业务流程没什么问题了。但是一个订单系统可不光是这么简单的一些业务而已

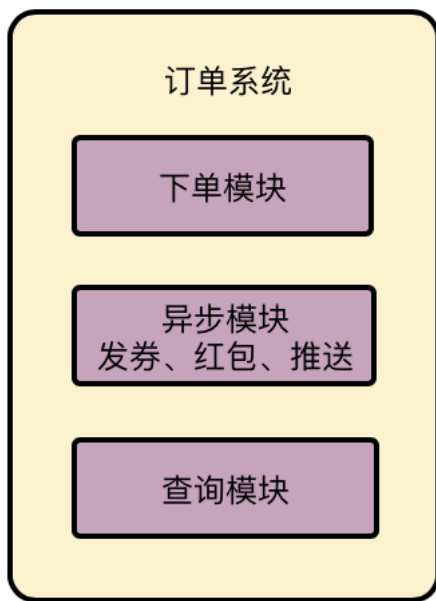
实际上订单系统在运行的过程中可能会跟底层的营销系统、购物车系统、商品系统、配送系统等大量的系统进行复杂的交互。当然，现在你不用一下子都搞的那么清楚，一步一步来。

现在给你讲讲订单系统在运行时的一些非核心业务流程。

首先订单系统在完成核心的下单业务流程之后，用户一定会查询自己的订单，那么订单系统务必要提供对应的**订单查询功能**。

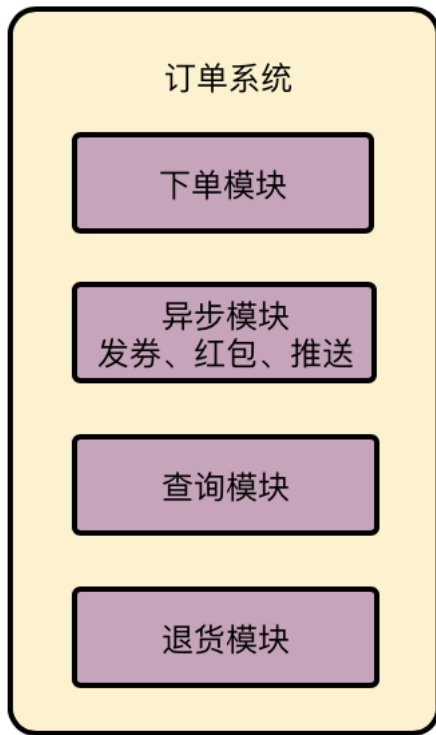
我们看下面的图，在这个图里画出了订单系统需要具备的一个功能模块

下单模块主要是用于创建订单，异步模块主要是在支付成功之后发优惠券、红包和推送，查询模块主要是提供订单的查询。



另外，当用户查询到一个订单列表之后，有时肯定会因为各种原因想要退货，这个是不可避免的。

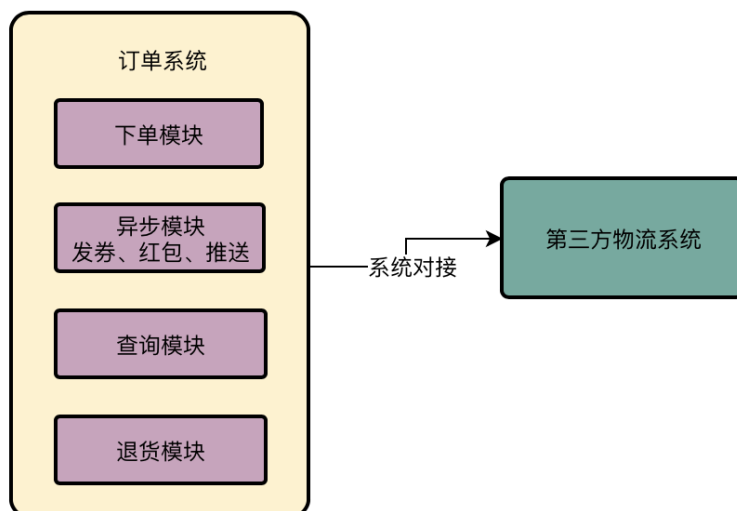
因此在订单系统中也得提供退货功能，在下面的图中我们加入退货模块。



此外，订单系统除了自己要提供的功能模块之外，还需要跟公司以外的第三方公司的一些系统进行对接。

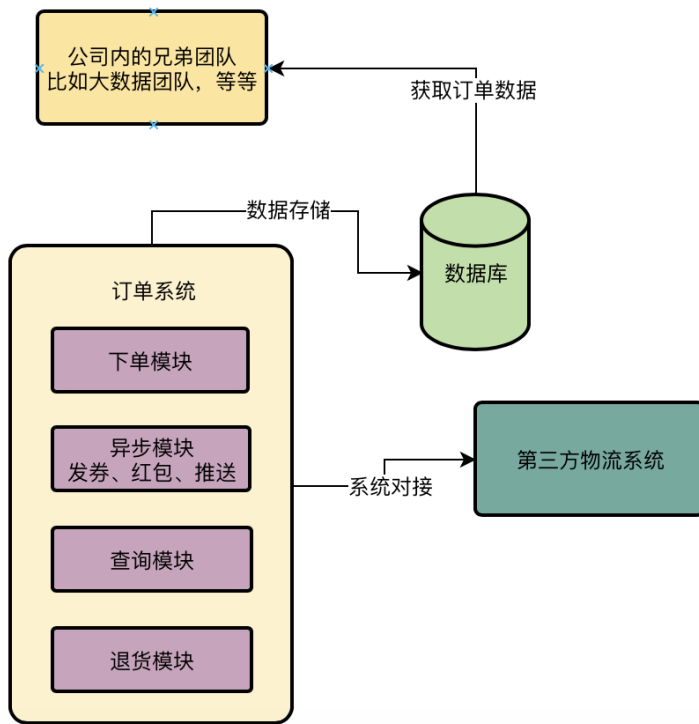
比如你想要查看订单的配送状态，那么就需要订单系统从第三方物流公司的系统中进行查询，才能让你看到。

因此在下面的图中，我们还得加入订单系统跟第三方系统的对接。



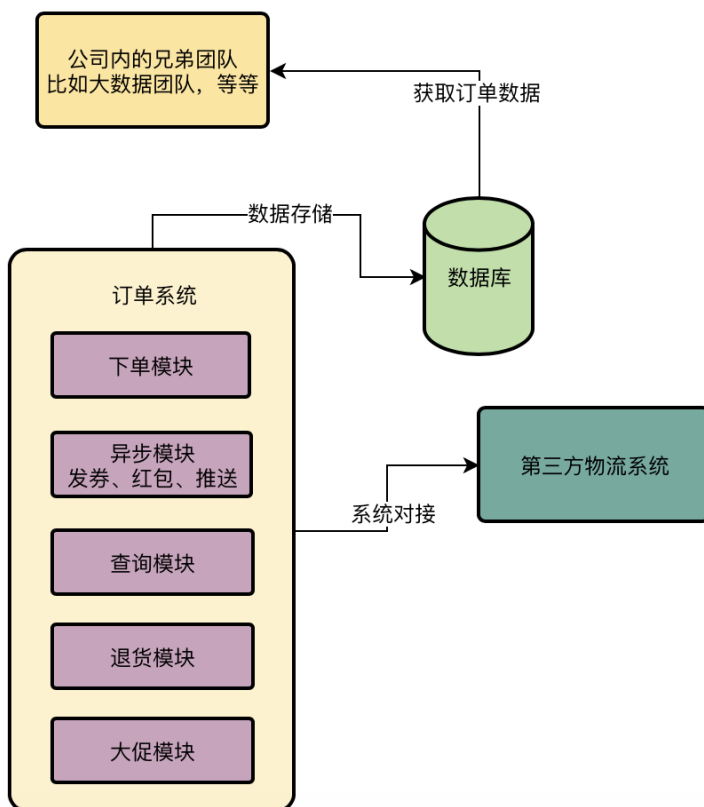
然后你应该知道，订单数据是一个公司的核心数据，很多时候公司内部的其他团队，比如大数据团队可能就需要获取订单数据进行分析，然后提供交易数据报表给公司的高层领导去看。

所以下面的图里，我画出来了订单的数据库以及第三方团队需要获取订单数据的一个业务场景。



最后就是在类似双11、秒杀等大促场景下，可能大量的用户会蹲点守在手机前，等待一些特价促销的商品开卖之后进行抢购，此时可能会对订单系统会产生一个流量洪峰，甚至影响正常的一些下单功能。

因此对于订单系统，往往要提供一个专门用于抗双11、秒杀等活动的大促模块，专门用于处理特殊活动下的高并发下单场景，这个模块也得加上：



讲到这里，明哥再次休息了一下，问道：

怎么样小猛，订单系统包含的一些功能模块，核心业务流程，以及一些非核心的业务流程和功能模块，大体都了解清楚了吗？

这些东西可是重中之重，因为后续我们要对订单系统进行架构改造，需要引入包括MQ、Elasticsearch、分库分表等大量的技术。

略微停顿了一下，明哥继续说道：当然，对你而言，我主要会安排你负责一部分基于MQ（消息中间件）技术的订单架构改造，暂时不会让你涉及其他的一些技术。

而对于MQ技术，几乎在上面这张图里所有的功能模块，都需要用MQ技术进行一定的架构改造。

小猛：明哥，你放心，听你讲完这套订单系统的业务流程和整体架构之后，我觉得非常的有意思，而且基本全部都听明白了，也一定会竭尽所能去学习MQ技术的相关知识，去做好我负责的那部分架构改造的！

明哥：好小伙子，态度很不错，但是可别高兴太早了，下面我来给你说说咱们的订单系统的线上环境的生产负载情况。要知道，我们公司可是有一定的业务量的，系统的整体负载和压力可不小，要做好这个事，还是有一定的难度的！

4、订单系统的真实生产负载情况

明哥：首先给你说一下，咱们公司是一个新兴的互联网公司，从创立到现在发展也没几年

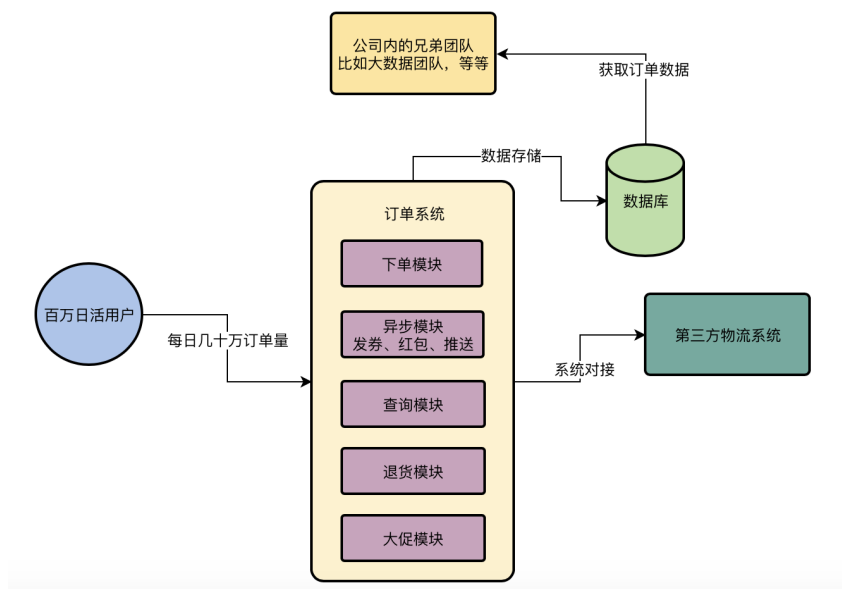
但是在我们公司的这个垂直电商领域里，我们算是这个小领域的NO.1，做的非常不错。到目前为止，我们公司大概积累的注册用户有几千万的数量。

当然，不是每个注册用户都会天天来逛你的APP。真的天天逛你的APP的，基本上得是死忠粉丝了。大部分人也就是每隔几天会来APP里逛一逛，可能有时候会下一个订单。

所以现在根据统计来看，每天我们APP活跃的用户数量是一两百万的样子，这个其实也不算少了，虽说我们是创业公司，但是也达到百万日活的体量了。然后每天新增的订单数量，目前大概是几十万的样子。

我估计随着公司发展，可能很快就會达到每日百万量级的订单数量，当然，在一些双11、618等大促活动的时候，我们现在的订单系统就可以达到单日百万订单的量级了，所以我们的系统压力还是比较大的。

因此我们先在下面的图中加入订单系统当前每日的订单数量。

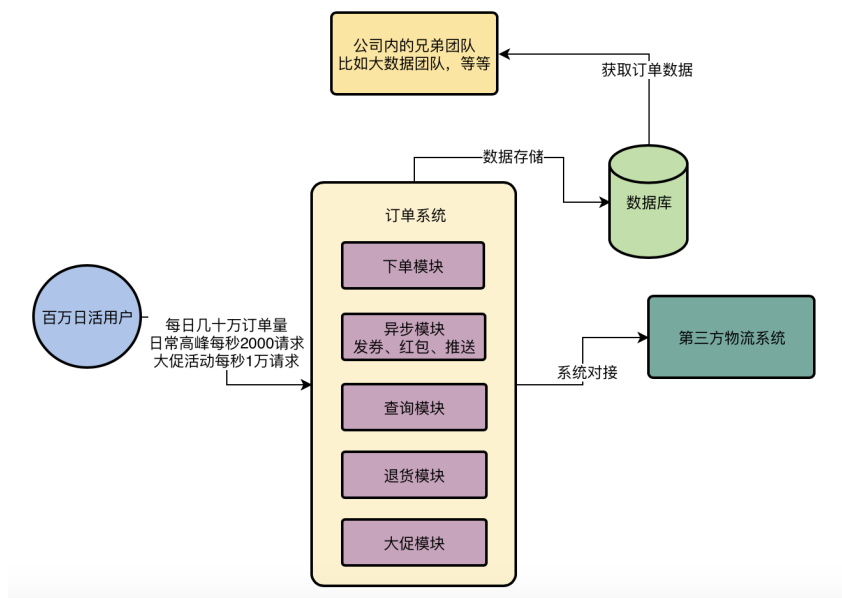


再说这个QPS，也就是**系统每秒的查询数量**，这个指标是说订单系统所有的核心以及非核心功能模块加起来，每秒钟有多少请求量。

对我们来说，在往常每天的高峰期，大概最多会达到每秒2000左右的访问量，不算太大。

但是如果要是那种特价商品限时秒杀的活动，那可能就会达到每秒1万以上的访问压力了。

因此我们继续在下面的图中，加入系统每秒的请求压力。



这就是我们这个订单系统的整体压力了，你可以看到，压力主要在两方面：

- 一方面是订单系统日益增长的数据量
- 一方面是在大促活动时每秒上万的访问压力

因此我们当前订单系统的整体架构是比较简单的，我们仅仅是让开发好的系统直接连接一台数据库服务器，所有的数据都是存储在里面的。

然后也是由这台数据库服务器去抗所有的访问压力，所以现在订单系统经常会在一些大促活动的时候出现不稳定的情况。

因为随着数据库中的订单数据越来越多，数据库的读写性能就会越来越差，尤其在大促活动高峰期的时候，数据库访问压力剧增，读写性能会进一步下降，经常出现请求过慢，请求超时等问题。

所以，咱们团队的任务，就是要尽快在订单系统的架构中引入更多的技术，进行大量的架构优化，让我们的订单系统逐步逐步的趋向于稳定。

小猛在听完明哥今天的讲解之后，认真的做了大量的笔记，打算今晚回家之后再反复消化，对系统的情况尽快熟悉了解。

一想到自己刚毕业就可以进入一个创业型互联网公司的核心技术团队，对有挑战的一个系统去做大量的优化工作，让系统变得越来越稳定，就觉得非常的兴奋和憧憬。

小猛对明哥说：放心吧老大，我一定会全力以赴的！

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

狸猫技术窝其他精品专栏推荐：

[《从零开始带你成为JVM实战高手》](#)

[《21天Java 面试突击训练营》（分布式篇）](#)（现更名为：[互联网Java工程师面试突击第2季](#)）

[互联网Java工程师面试突击（第1季）](#)

重要说明：

如何提问：每篇文章都有评论区，大家可以尽情在评论区留言提问，我会逐一答疑

如何加群：购买了狸猫技术窝专栏的小伙伴都可以加入**狸猫技术交流群**

具体加群方式，请参见[目录菜单](#)下的文档：《付费用户如何加群？》（[购买后](#)可见）